

Knowledge Distillation: Evolution, Methodologies, and Emerging Applications in Large Language Models

Pranav Shaji*, Ms. Riya K Prakash**, Prof (Dr). L C Manikandan***

*B.Tech. Student, Department of CSE, Universal Engineering College, Thrissur, Kerala, India

Email: pranavshajiofficial@gmail.com

**Assistant Professor, Department of CSE, Universal Engineering College, Thrissur, Kerala, India

Email: riyajyothish13@gmail.com

***Professor, Department of CSE, Universal Engineering College, Thrissur, Kerala, India

Email: lcmanikandan@gmail.com

Abstract:

Knowledge Distillation (KD) is a technique for transferring knowledge in which a smaller student model is trained using the information generated by a larger or more capable teacher model. The main aim is to create models that need less memory and computational power while at the same time preserving as much of the teacher's predictive power as possible. Starting from the early version of the teacher-student approach that relied on softened output distributions, KD has evolved into a wide range of methods that transfer responses, intermediate representations, relationships, attention information, or task-specific capabilities. More recent research has applied these ideas to Transformer encoders and Large Language Models (LLMs), since the sequential and generative nature of language modelling brings with it challenges which do not exist in the case of traditional classification. This paper looks at the basic principles of KD, its main methodological categories, some typical modern techniques, its use with Transformer and LLM models, and how it interacts with quantization. A comparison based on the existing literature is made in order to see how various approaches balance model size, computational efficiency, and the amount of capability retained. The review also points out some unresolved problems, such as a mismatch between the capacity of the teacher and student models, differences in architecture, knowledge loss, and the preservation of reasoning abilities in smaller language models.

Keywords — Knowledge Distillation, Model Compression, Teacher–Student Learning, Autoregressive Generation, Transformer, Large Language Model, Small Language Model, Quantization.

I. INTRODUCTION

Current deep learning systems increasingly depend on models with large numbers of parameters and substantial training. Increasing model capacity can improve performance, but the resulting systems may be difficult to deploy when memory, latency, or hardware availability is limited. The original motivation for knowledge distillation was related to this deployment problem: a cumbersome model or ensemble could first be trained for high predictive performance, after which its useful behaviour could

be transferred to a smaller model intended for practical use [1].

Knowledge Distillation treats the knowledge contained in a trained model as something that can be transferred independently of the exact parameterization of that model. Instead of requiring the student to learn only from the ground-truth label, the student can also learn from the teacher's output distribution. The relative probabilities assigned to alternative classes can contain information about similarities and confusions that a one-hot label does

not express [1]. This was what led to the foundation for later KD research.

The field then subsequently expanded beyond output probabilities. Surveys of KD classify methods according to the type of knowledge transferred, the source of that knowledge, the training arrangement, and the relationship between teacher and student architectures [2], [3], [4]. Intermediate features, attention maps, pairwise relations, and other representations can be used as additional supervision. This allows distillation to be adapted to different architectures and tasks.

The emergence of Transformer-based models and LLMs has made KD particularly relevant to efficient natural language processing. Models such as BERT demonstrated the usefulness of transferring information from a large pretrained language model into a compact student, while later work researched on distillation for generative LLMs [5], [6], [7]. At the same time, KD can be combined with other efficiency techniques. Quantization reduces the numerical precision used to represent model parameters and activations, while KD can provide additional training signals to recover some of the accuracy lost through aggressive compression [8].

This paper reviews the development of KD from its original formulation to current applications in Transformers and LLMs. The discussion focuses on the basic teacher-student mechanism, major categories of transferable knowledge, representative advanced approaches, LLM-specific distillation, and KD combined with quantization. The objective is to present a connected view of the field rather than treating individual methods as isolated techniques.

II. KNOWLEDGE DISTILLATION: FUNDAMENTALS AND METHODOLOGY

A. Teacher-Student Framework

A standard KD system contains two models. The teacher is generally larger or more capable and is assumed to have learned a useful representation of the task. The student is smaller and is trained to approximate selected information from the teacher. The teacher is normally trained first and kept fixed during the student's training in an offline distillation setting [1], [2].

For a classification task, the teacher receives an input x and produces logits z_T . A softmax function

converts these logits into probabilities. The student produces its own logits z_S and probability distribution. The student objective can combine ordinary supervision from the ground-truth labels with a distillation objective that encourages the student distribution to resemble the teacher distribution.

The framework therefore separates two forms of supervision. Ground-truth labels tell the student which answer is correct, whereas the teacher provides a richer description of how the model distributes probability among possible alternatives. This additional information is one of the central ideas behind KD [1].

B. Hard Targets and Soft Targets

A conventional classification target is often represented as a one-hot vector. For example, if an image belongs to a flower, the target may assign probability one to that flower and zero to every other classification. Such a representation does not give a correct indication of which other flower could be more plausible.

A teacher's output can instead be represented as a soft probability distribution. If the teacher assigns relatively high probability to flower and smaller but different probabilities to tree and leaves, these values provide information about the teacher's learned similarity structure. Hinton et al. argued that these relative probabilities can reveal useful information about generalization and can therefore serve as training targets for the student [1].

Soft targets are especially useful when the teacher is much more capable than the student. They allow the student to observe not only the final decision but also the teacher's confidence in the certainty. This principle motivated many subsequent response-based KD methods.

C. Temperature Scaling

When a normal softmax is highly confident, the largest probability may dominate the remaining values. To expose more information in the distribution, KD commonly introduces a temperature parameter T :

$$p_i(T) = \exp(z_i / T) / \sum_j \exp(z_j / T) \quad (1)$$

A larger temperature produces a softer distribution. During distillation, the teacher and student distributions are evaluated using the same temperature so that the student learns to reproduce the softened teacher behaviour. Hinton et al. also showed that the contribution of the distillation loss should be scaled appropriately when temperature is used [1].

Temperature should therefore not be understood as merely a numerical trick. Its purpose is to control how much information about non-dominant classes remains visible to the student.

D. Knowledge Distillation Loss

A common objective combines ordinary supervised learning and teacher supervision:

$$L = \alpha L_{\text{soft}} + (1 - \alpha) L_{\text{hard}} \quad (2)$$

Where L_{hard} represents the loss against the ground-truth label, L_{soft} measures disagreement between teacher and student outputs, and α controls the relative contribution of the two signals. The KD component is commonly based on a divergence between softened teacher and student distributions.

The exact loss is not universal. One of the major themes in KD research is that the choice of knowledge source and objective can strongly affect the outcome. Later methods therefore modify the distillation objective or introduce additional losses for intermediate representations and relationships [2], [9].

E. Types of Knowledge

1) Response-Based Distillation

Response-based KD uses the outputs of the teacher as the primary knowledge source. The classical soft-label formulation belongs to this category. Its major advantage is that the teacher and student do not necessarily need identical internal architectures because only their outputs need to be compared [2].

2) Feature-Based Distillation

Feature-based KD transfers information from intermediate layers. Instead of checking only whether the final predictions agree, the student is encouraged to reproduce selected internal representations of the teacher. FitNets introduced the

idea of using intermediate teacher representations, or hints, to guide thinner networks [10]. In Transformer models, hidden states and layer outputs can similarly act as distillation targets [5], [11].

The benefit is that intermediate representations can expose information that is lost before the final prediction. The main difficulty is that teacher and student feature dimensions or layer structures may differ, requiring alignment mechanisms or projection layers.

3) Relation-Based Distillation

Relation-based KD focuses on relationships among representations rather than matching individual features directly. For example, a student may be trained to preserve distances, angles, or relationships between samples in the teacher's representation space. Relational Knowledge Distillation demonstrated that such structural information can be useful when transferring knowledge [12].

This perspective is important because knowledge is not necessarily contained in isolated activations. The organization of representations can itself encode information about the task.

F. Distillation Training Schemes

1) Offline Distillation

In offline KD, the teacher is trained beforehand and its parameters remain fixed while the student is optimized. This is conceptually simple and allows a high-capacity teacher to be used solely as a source of supervision [2].

2) Online Distillation

Online KD trains teacher-like and student-like models together or constructs the teaching signal dynamically during training. This can reduce dependence on a separately pretrained teacher but may introduce additional optimization complexity. Survey literature places online methods among the broader training-scheme choices in KD [2], [4].

3) Self-Distillation

Self-distillation removes the strict requirement for a separate larger teacher. Knowledge from another stage, layer, branch, or version of the same model can be used as a training signal. This approach demonstrates that the useful regularization effect of

KD is not necessarily limited to a large-to-small model relationship [2], [3].

III. EVOLUTION AND ADVANCED KNOWLEDGE DISTILLATION

A. From Classical KD to Modern KD

The original formulation established output-level transfer as a practical mechanism for model compression. Subsequent research explored how the choice of knowledge representation affects transfer. Feature-based and relation-based methods attempt to provide supervision at deeper levels, while self-distillation and multi-teacher approaches modify the source and organization of knowledge [2], [3].

The broader evolution can be summarized as a movement from asking the student to imitate a teacher's final answer toward asking it to reproduce selected aspects of the teacher's internal behaviour. Recent surveys also show the expansion of KD into multimodal learning, foundation models, heterogeneous architectures, and generative models [4].

B. Decoupled Knowledge Distillation

Decoupled Knowledge Distillation (DKD) revisits the information contained in the teacher's output distribution. Instead of treating the complete soft distribution as a single indivisible target, DKD separates the contribution associated with the target class from information associated with non-target classes [9].

The motivation is that the two components play different roles. Target-class information is strongly related to whether the student recognizes the correct class, while the distribution over non-target classes contains information about inter-class relationships. Separating these components gives the training process independent control over their contributions.

DKD therefore illustrates an important pattern in modern KD research: rather than simply increasing model size or adding more supervision, researchers can improve distillation by analysing what information is actually being transferred and by redesigning the objective around that structure [9].

C. Heterogeneous Architecture Distillation

Traditional KD is usually easy when teacher and student have similar structures. However, practical compression frequently requires changing the architecture itself. A student may use a different network family, number of layers, or computational organization.

One-for-All investigates this heterogeneous setting and aims to reduce the architectural restrictions that make direct knowledge transfer difficult [13]. This direction is important for deployment because the smallest or fastest architecture is not always a reduced version of the teacher. The ability to transfer knowledge across heterogeneous models therefore broadens the design space available to system developers.

D. Multi-Teacher Distillation

A single teacher does not necessarily contain all useful information. Multi-teacher approaches use several teachers so that the student can learn complementary knowledge. Different teachers may specialize in different representations, tasks, or capabilities. The main challenge is deciding how the teacher signals should be combined without creating conflicting supervision [2], [4].

IV KNOWLEDGE DISTILLATION FOR TRANSFORMERS AND LARGE LANGUAGE MODELS

A. Motivation for Transformer Distillation

Transformer models changed the requirements for KD because useful knowledge is distributed across multiple attention and feed-forward layers. In an encoder such as BERT, the final output alone may not represent all of the information learned by the teacher. Distillation can therefore target embeddings, hidden states, attention information, and output predictions [5].

B. BERT and Transformer Distillation

TinyBERT is a representative approach to Transformer compression. It performs a two-stage distillation strategy and transfers knowledge from multiple components of the BERT teacher, including embedding information, Transformer-layer representations, attention-related information, and prediction outputs [5].

This is a significant transition from classical response-based KD. Instead of treating the teacher as a black-box classifier, the student is trained to reproduce selected internal properties of the Transformer. TinyBERT demonstrated that substantial reductions in model size and inference cost can be achieved while retaining strong natural-language understanding performance [5].

Other approaches followed similar principles. Patient Knowledge Distillation uses intermediate representations from selected teacher layers to guide the student, while DistilBERT applies a compact Transformer architecture with a distillation objective to reduce the cost of BERT-like models [11], [14].

C. Why LLM Distillation Is Different

Distilling a generative LLM is more difficult than distilling a conventional classifier. A classifier typically produces a probability distribution over a fixed set of classes. An autoregressive language model instead predicts a sequence token by token. Each generated token changes the context used to predict later tokens.

This creates several differences. First, the student must learn a conditional distribution over possible continuations. Second, the quality of a generated response depends on a sequence of interdependent predictions. Third, the teacher can express behaviours such as instruction following, style, factual patterns, and reasoning strategies that are not captured by a single classification output [6].

LLM KD can therefore use several forms of teacher information. A white-box setting may expose logits or internal states, whereas a black-box setting may rely primarily on teacher-generated responses. The latter is particularly important when the teacher's parameters are inaccessible [6].

D. White-Box and Black-Box LLM Distillation

White-box distillation has access to internal teacher information such as logits, hidden representations, or attention-related signals. This provides rich supervision but requires access to the teacher model or its inference internals.

Black-box distillation instead treats the teacher as a response generator. Prompts are supplied to the teacher, generated outputs are collected, and the resulting examples are used to train the student. This can make distillation possible even when only an

API or externally hosted model is available. The trade-off is that the student has less direct access to the teacher's internal probability distribution and must infer capabilities from generated examples [6].

E. MiniLLM and Generative Distillation

MiniLLM studies the distillation of generative language models and argues that directly applying conventional KD objectives to autoregressive generation can be problematic. The work explores reverse-KL-based optimization so that the smaller model is encouraged to learn the teacher's high-probability regions in a way better suited to generation [7].

The importance of MiniLLM is not limited to one loss function. It demonstrates a broader principle: when the behaviour being transferred changes, the distillation objective may also need to change. A method designed for fixed-label classification cannot automatically be assumed to be optimal for open-ended language generation.

The LLM KD literature therefore increasingly considers not only compression but also capability transfer. Surveys identify areas such as instruction following, reasoning, domain knowledge, alignment-related behaviour, and synthetic-data generation as active topics [6].

V. KNOWLEDGE DISTILLATION FOR EFFICIENT ARTIFICIAL INTELLIGENCE

A. Model Compression as a Combination of Techniques

KD is one component of a broader model-efficiency toolbox. Pruning removes parameters or structures, quantization reduces numerical precision, and architectural compression reduces depth or width. KD is different because it primarily changes the training signal: it teaches a model to imitate information from another model.

These techniques can be combined. For example, a student can be trained using KD and subsequently quantized, or KD can be integrated directly into quantization-aware training. The latter is particularly useful when aggressive quantization introduces a significant accuracy loss.

B. Knowledge Distillation and Quantization

Quantization represents model values using fewer bits than conventional floating-point representations.

Lower precision can reduce memory consumption and potentially improve inference efficiency, but aggressive quantization can damage model accuracy.

KD provides an additional source of information that can help the compressed model recover useful behaviour. Instead of learning only from ground-truth labels while its weights are constrained to low precision, the quantized student can also be guided by a higher-precision teacher.

C. Quantization-Aware Knowledge Distillation

Kim et al. researched KD for quantization-aware training of large Transformer encoders. Their analysis focused on attention recovery and examined which intermediate knowledge sources and loss functions are useful when Transformer weights are quantized aggressively [8].

The study found that intermediate-layer KD can be important for quantization-aware training and investigated losses based on attention maps and attention outputs. The work also showed that the best choice of knowledge source can depend on the model and task. This is an important observation because it indicates that there is no universally optimal KD signal for every compression setting [8].

The relationship can therefore be expressed conceptually as follows:

Large Teacher → Knowledge Signal → Quantized Student

Here, quantization controls how efficiently the student is represented, while KD controls how the student is trained to preserve useful behaviour.

D. Edge and Resource-Constrained Deployment

The combination of KD and quantization is particularly relevant to edge computing and other resource-constrained environments. Smaller models can reduce memory requirements and latency, while low-precision representations can further reduce storage and computational cost. For applications where cloud-scale models cannot be deployed locally, these techniques provide a route toward capable local inference.

This direction is also relevant to Small Language Models. Instead of attempting to reproduce the entire capability of a large model, a student can be optimized for a specific task or set of capabilities and

then compressed further for deployment on constrained hardware.

VI ANALYSIS

This section presents a literature-based comparison rather than results from a newly conducted experiment. Representative knowledge distillation methods are shown in Table I.

TABLE I

Representative Knowledge Distillation Methods

Method	Domain	Teacher–Student	Main Idea
Classical KD	Vision / classification	Large model → small model	Transfer softened output distributions
FitNets	Vision	Wide/deep teacher → thin student	Use intermediate representations as hints
RKD	Vision	Teacher → student	Transfer relationships between representations
DKD	Vision	Teacher → student	Separate target and non-target knowledge
TinyBERT	NLP	BERT → compact Transformer	Distill multiple Transformer components
MiniLLM	LLM	LLM → smaller LM	Adapt KD to autoregressive generation
KD + QAT	NLP / Transformer	Full precision → quantized model	Use KD to recover performance under aggressive quantization

A. Efficiency and Performance Trade-offs

The literature shows that model compression is inherently a trade-off. A smaller student normally reduces memory and inference cost, but an overly small student may lack the capacity to reproduce all teacher behaviours. KD can improve this trade-off by supplying richer supervision than hard labels alone [1], [2].

Transformer-specific methods demonstrate that retaining performance may require more than output matching. TinyBERT transfers information from several levels of the teacher and reports strong performance despite substantial model reduction [5]. Quantization-aware studies further show that the choice of intermediate signal matters when numerical precision becomes very low [8].

B. Classical KD versus LLM KD

Classical KD generally assumes a fixed prediction space, whereas LLM KD operates over sequences whose predictions depend on previous generated tokens. As a result, the teacher's knowledge may be expressed through token distributions, generated demonstrations, hidden representations, or higher-level capabilities.

The LLM literature consequently contains both white-box and black-box approaches and increasingly focuses on capability-specific transfer. MiniLLM illustrates how the objective itself may need to be redesigned for generative modelling [6], [7].

C. Key Observations

Four observations emerge from the reviewed literature. First, KD has moved from output imitation toward richer forms of knowledge transfer. Second, the appropriate knowledge source depends on the architecture and task. Third, Transformer and LLM distillation require methods that account for internal representations and autoregressive generation. Fourth, KD is increasingly being combined with other efficiency techniques rather than being treated as an isolated compression method [3], [4].

VII CHALLENGES AND FUTURE DIRECTIONS**A. Teacher-Student Capacity Gap**

A student with substantially lower capacity cannot reproduce every aspect of a large teacher. The challenge is therefore to identify which knowledge is most valuable for the target application and transfer that knowledge efficiently. This problem becomes more significant when a small language model is expected to reproduce several capabilities of a much larger model.

B. Knowledge Loss During Distillation

Distillation is necessarily selective. The student does not receive the teacher's parameters directly and may not have enough capacity to encode every useful behaviour. Choosing the correct knowledge representation and loss is therefore central to successful KD [2], [4].

C. Architecture Mismatch

Different teacher and student architectures may represent information in incompatible forms. Feature matching becomes difficult when layer dimensions, attention structures, or model depths differ. Heterogeneous KD is an active direction because it could allow practitioners to choose students based on deployment requirements rather than architectural similarity [13].

D. Distilling Reasoning and Emergent Capabilities

For LLMs, matching ordinary token predictions may not be sufficient to preserve complex capabilities such as multi-step reasoning. A student may imitate the final answer while failing to reproduce the process that generated it. Future KD research therefore needs to investigate how reasoning traces, intermediate representations, demonstrations, and capability-specific supervision should be transferred [6].

E. KD with Quantization and Other Compression Methods

Combining KD with quantization, pruning, low-rank adaptation, or architecture reduction creates a larger design space but also makes optimization more complex. The optimal KD signal may change as the student becomes more aggressively compressed. Quantization-aware KD studies already show that attention-related supervision can behave differently under low-bit constraints [8].

F. Future Research Opportunities

Several directions appear promising. Adaptive distillation could select different knowledge sources depending on the student's capacity. Architecture-independent KD could allow arbitrary teacher and student combinations. Capability-aware LLM distillation could target particular skills instead of attempting to copy the complete teacher. Another promising direction is joint optimization of KD with quantization so that the student is trained for its final deployment constraints from the beginning.

For practical Small Language Models, a particularly relevant research question is how much of a large model's capability can be retained when the student is simultaneously reduced in parameter count and numerical precision. Such a question

naturally connects KD, quantization, and efficient local AI deployment.

VIII CONCLUSION

Knowledge Distillation has developed from a relatively simple teacher-student compression technique into a broad framework for transferring useful information between models. The original use of softened predictions demonstrated that a large model contains information beyond hard labels. Later research expanded the transferable knowledge to intermediate representations, relationships, attention information, and other model behaviours.

The transition to Transformer and LLM architectures has introduced new directions in the field. Transformer distillation can benefit from internal-layer supervision, while LLM distillation must account for autoregressive generation and the transfer of higher-level capabilities. Methods such as TinyBERT and MiniLLM illustrate how distillation objectives can be adapted to these settings.

KD also has an important role in efficient AI when combined with quantization. In this setting, quantization reduces representation cost while KD supplies additional supervision for recovering useful behaviour. The reviewed literature suggests that future progress will depend less on a single universal KD method and more on adaptive approaches that select suitable knowledge sources, objectives, and compression strategies for a particular teacher, student, task, and deployment environment.

REFERENCES

- [1] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.
- [2] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, pp. 1789–1819, 2021.
- [3] A. Moslemi and A. Briskina, "A survey on knowledge distillation: Recent advancements," *Machine Learning with Applications*, 2024.
- [4] A. M. Mansourian et al., "A comprehensive survey on knowledge distillation," *Transactions on Machine Learning Research*, 2025.
- [5] X. Jiao et al., "Tinybert: Distilling bert for natural language understanding," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020, pp. 4163–4174.
- [6] X. Xu et al., "A survey on knowledge distillation of large language models," *arXiv preprint arXiv:2402.13116*, 2024.
- [7] Y. Gu, L. Dong, F. Wei, and M. Huang, "Minillm: Knowledge distillation of large language models," *arXiv preprint arXiv:2306.08543*, 2023.
- [8] M. Kim, S. Lee, S. Hong, D.-S. Chang, and J. Choi, "Understanding and improving knowledge distillation for quantization-aware training of large transformer encoders," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022, pp. 6713–6725.
- [9] B. Zhao, Q. Cui, R. Song, Y. Qiu, and J. Liang, "Decoupled knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11953–11962.
- [10] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," in *International Conference on Learning Representations*, 2015.
- [11] S. Sun, Y. Cheng, Z. Gan, and J. Liu, "Patient knowledge distillation for bert model compression," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019, pp. 4323–4332.
- [12] W. Park, D. Kim, Y. Lu, and M. Cho, "Relational knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3967–3976.
- [13] [1] Z. Hao et al., "One-for-all: Bridge the gap between heterogeneous architectures in knowledge distillation," in *Advances in Neural Information Processing Systems*, 2023.
- [14] [2] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.